

10/30/00
JC951 U.S. PTO

10-31-00

A

PATENT

Attorney Docket No. HAYSCHR.002A

Date: October 30, 2000

Page 1

JC951 U.S. PTO
09/699798

10/30/00

ASSISTANT COMMISSIONER FOR PATENTS

WASHINGTON, D.C. 20231

ATTENTION: BOX PATENT APPLICATION

Sir:

Transmitted herewith for filing is the patent application of

Inventor(s): Timothy J. Smith and Brad Silen

For: **METHODS AND APPARATUSES FOR SYNCHRONIZING MIXED-MEDIA DATA FILES**

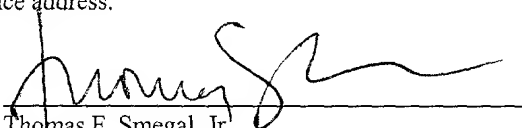
Enclosed are:

- (X) 14 sheet(s) of drawing.
- (X) Three verified statements to establish small entity status under 37 CFR 1.9 and 37 CFR 1.27.
- (X) Initial signed Declaration and Power of Attorney by inventor(s).
- (X) Return prepaid postcard.

CLAIMS AS FILED

FOR	NUMBER FILED	NUMBER EXTRA	RATE	FEE
Basic Fee			\$355	\$355
Total Claims	31 - 20 =	11 ×	\$9	\$99
Independent Claims	7 - 3 =	4 ×	\$40	\$160
If application contains any multiple dependent claims(s), then add			\$135	\$0
TOTAL FILING FEE				\$614

- (X) A check in the amount of \$614.00 to cover the filing fee is enclosed.
- (X) The Commissioner is hereby authorized to charge any additional fees which may be required, now or in the future, or credit any overpayment to Account No. 11-1410. A duplicate copy of this sheet is enclosed.
- (X) Please use Customer No. 20,995 for the correspondence address.


Thomas F. Smegal, Jr.
Registration No. 20,732
Attorney of Record

W \DOCS\TFS\TFS-6156.DOC
102700

KNOBBE, MARTENS, OLSON & BEAR

A LIMITED LIABILITY PARTNERSHIP INCLUDING
PROFESSIONAL CORPORATIONS

PATENT, TRADEMARK AND COPYRIGHT CAUSES

201 CALIFORNIA STREET

SUITE 1150

SAN FRANCISCO, CALIFORNIA 94111-3335

(415) 954-4114

FAX (415) 954-4111

INTERNET WWW.KNOB.COM

JOSEPH S. CIANFRANI
JOSEPH M. REISMAN, PH.D.
WILLIAM R. ZIMMERMAN
GLEN L. NUTTALL
ERIC S. FURMAN, PH.D.
TIRZAH ABÉ LOWE
GEOFFREY Y. IIDA
ALEXANDER S. FRANCO
SANJIVPAL S. GILL
SUSAN M. MOSS
JAMES W. HILL, M.D.
ROSE M. THIESSEN, PH.D.
MICHAEL L. FULLER
MICHAEL A. GUILIANA
MARK J. KERTZ
RABINDER N. NARULA
BRUCE S. ITCHKAWITZ, PH.D.
PETER M. MIDGLEY
THOMAS S. MCCLENAHAN
MICHAEL S. OKAMOTO
JOHN M. GROVER
MALLARY K. DE MERLIER
IRFAN A. LATEEF
AMY C. CHRISTENSEN
SHARON S. NG
MARK J. GALLAGHER, PH.D.
DAVID G. JANKOWSKI, PH.D.
BRIAN C. HORNE
PAYSON J. LEMEILLEUR
DIANA W. PRINCE
SHEILA N. SWAROOP

OF COUNSEL

JERRY R. SEILER
PAUL C. STEINHARDT

JAPANESE PATENT ATTY
KATSUHIRO ARAI**

EUROPEAN PATENT ATTY
MARTIN HELLEBRANDT*

KOREAN PATENT ATTY
MINCHEOL KIM

SCIENTISTS & ENGINEERS
(NON-LAWYERS)

RAYMOND J. SALENIKS**
DANIEL E. JOHNSON, PH.D.**
JEFFERY KOEPEKE, PH.D.**
KHURRAM RAHMAN, PH.D.
JENNIFER A. HAYNES, PH.D.
BRENDAN P. O'NEILL, PH.D.
THOMAS Y. NAGATA
YASHWANT VAISHNAV, PH.D.
MEGUMI TANAKA
CHE S. CHERESKIN, PH.D.**
ERIK W. ARCHBOLD
PHILIP C. HARTSTEIN
JULIE A. HOPPER
CHRIS S. CASTLE
JAMES W. AUSLEY
R. P. CARON, PH.D.
JENNIFER HAYES
KIRK E. PASTORIAN, PH.D.
CHARLES T. RIDGELY
KEITH R. MCCOLLUM
BONNY YEUNG, PH.D.
CONNIE C. TONG, PH.D.

* A PROFESSIONAL CORPORATION
* ALSO BARRISTER AT LAW (U.K.)
** U.S. PATENT AGENT

LOUIS J. KNOBBE*
DON W. MARTENS*
GORDON H. OLSON*
JAMES B. BEAR
DARRELL L. OLSON*
WILLIAM B. BUNKER
WILLIAM H. NIEMAN
ARTHUR S. ROSE
JAMES F. LESNIAK
NED A. ISRAELSEN
DREW S. HAMILTON
JERRY T. SEWELL
JOHN B. SGANGA, JR.
EDWARD A. SCHLATTER
GERARD VON HOFFMANN
JOSEPH R. RE
CATHERINE J. HOLLAND
JOHN M. CARSON
KAREN VOGEL WEIL
ANDREW H. SIMPSON
JEFFREY L. VAN HOESEAR
DANIEL E. ALTMAN
MARGUERITE L. GUNN
STEPHEN C. JENSEN
VITO A. CANUSO III
WILLIAM H. SHREVE
LYNDA J. ZADRA-SYMES†
STEVEN J. NATAUPSKY
PAUL A. STEWART
JOSEPH F. JENNINGS
CRAIG S. SUMMERS
ANNEMARIE KAISER
BRENTON R. BABCOCK

THOMAS F. SMEGAL, JR.
MICHAEL H. TRENMOLM
DIANE M. REED
JONATHAN A. BARNEY
RONALD J. SCHOENBAUM
JOHN R. KING
FREDERICK S. BERRETTA
NANCY WAYS VENSKO
JOHN P. GIEZENTANNER
ADEEL S. AKHTAR
GINGER R. DREGER
THOMAS R. ARNO
DAVID N. WEISS
DANIEL HART, PH.D.
DOUGLAS G. MUEHLHAUSER
LORI LEE YAMATO
MICHAEL K. FRIEDLAND
STEPHEN M. LOBBIN
STACEY R. HALPERN
DALE C. HUNT, PH.D.
LEE W. HENDERSON, PH.D.
DEBORAH S. SHEPHERD
RICHARD E. CAMPBELL
MARK M. ABUMERI
JON W. GURKA
ERIC M. NELSON
MARK R. BENEDICT, PH.D.
PAUL N. CONOVER
ROBERT J. ROBY
SABING H. LEE
KAROLINE A. DELANEY
JOHN W. HOLCOMB
JAMES J. MULLEN, III, PH.D.

Assistant Commissioner for Patents
Washington, D.C. 20231

CERTIFICATE OF MAILING BY "EXPRESS MAIL"

Attorney Docket No. : HAYSCHR.002A

Applicant(s) : Timothy J. Smith and Brad Silen

For: METHODS AND APPARATUSES FOR SYNCHRONIZING MIXED-MEDIA DATA FILES

Attorney : Thomas F. Smegal, Jr.

"Express Mail" Mailing Label No. : EL544355765US

Date of Deposit : October 30, 2000

I hereby certify that the accompanying

Transmittal in Duplicate; Specification in 48 pages; 14 sheets of drawings;
SIGNED Declaration and Power of Attorney in 2 pages; Three Small Entity
Statement(s); Check(s) for Filing Fee(s); Return Prepaid Postcard

are being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 CFR 1.10 on the date indicated above and are addressed to the Assistant Commissioner for Patents, Washington, D.C. 20231.


Jason Wrinkle



VERIFIED STATEMENT (DECLARATION) CLAIMING SMALL-ENTITY STATUS

I, the undersigned, do hereby declare that:

☒ I am an official of the small business concern identified below:

NAME OF CONCERN: James Allen Group

ADDRESS OF CONCERN: 1335 Cristina Avenue, San Jose, California 95125

I further declare that the above-identified small business concern qualifies as a small business concern as defined in 13 CFR 121.12, and reproduced in 37 CFR 1.9(d), for purposes of paying reduced fees to the United States Patent and Trademark Office, in that the number of employees of the concern, including those of its affiliates, does not exceed 500 persons. For purposes of this statement, (1) the number of employees of the business concern is the average over the previous fiscal year of the concern of the persons employed on a full-time, part-time or temporary basis during each of the pay periods of the fiscal year, and (2) concerns are affiliates of each other when either, directly or indirectly, one concern controls or has the power to control the other, or a third party or parties controls or has the power to control both. I further declare that rights under contract or law have been conveyed to and remain with the small business concern identified above with regard to the invention described in the patent or application identified above.

The individual, concern or organization identified above has not assigned, granted, conveyed or licensed, and is under no obligation under contract or law to assign, grant, convey or license, any rights in the invention to any person who would not qualify as an independent inventor under 37 CFR 1.9(c) if that person had made the invention, or to any concern which would not qualify as a small business concern under 37 CFR 1.9(d) or a nonprofit organization under 37 CFR 1.9(e).

If the rights held by the above-identified individual, concern or organization are not exclusive, each individual, concern or organization having rights in the invention are identified below. Each such individual, concern or organization must file separate verified statements averring to their status as small entities.

***NOTE: Separate verified statements are required from each named person, concern or organization having rights to the invention averring to their status as small entities. (37 CFR 1.27).**

FULL NAME: Timothy J. Smith

ADDRESS: 1335 Cristina Avenue, San Jose, California 95125

☒ INDIVIDUAL ☐ SMALL BUSINESS CONCERN ☐ NONPROFIT ORGANIZATION

FULL NAME: Brad Silen

ADDRESS: #5 South Forty Dock, Sausalito, CA 94965

☒ INDIVIDUAL ☐ SMALL BUSINESS CONCERN ☐ NONPROFIT ORGANIZATION

I acknowledge the duty to file, in this application or patent, notification of any change in status resulting in loss of entitlement to small-entity status prior to paying, or at the time of paying, the earliest of the issue fee or any maintenance fee due after the date on which status as a small entity is no longer appropriate. (37 CFR 1.28(b)).

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under section 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of the application, any patent issuing thereon, or any patent to which this verified statement is directed.

Parameter	Value	Parameter	Value	Parameter	Value	Parameter	Value
Standard error	0.0000	Standard error	0.0000	Standard error	0.0000	Standard error	0.0000
Mean	0.0000	Mean	0.0000	Mean	0.0000	Mean	0.0000
Standard deviation	0.0000	Standard deviation	0.0000	Standard deviation	0.0000	Standard deviation	0.0000
Minimum	0.0000	Minimum	0.0000	Minimum	0.0000	Minimum	0.0000
Maximum	0.0000	Maximum	0.0000	Maximum	0.0000	Maximum	0.0000
Skewness	0.0000	Skewness	0.0000	Skewness	0.0000	Skewness	0.0000
Kurtosis	0.0000	Kurtosis	0.0000	Kurtosis	0.0000	Kurtosis	0.0000
Alpha	0.0000	Alpha	0.0000	Alpha	0.0000	Alpha	0.0000
Beta	0.0000	Beta	0.0000	Beta	0.0000	Beta	0.0000
Gamma	0.0000	Gamma	0.0000	Gamma	0.0000	Gamma	0.0000
Delta	0.0000	Delta	0.0000	Delta	0.0000	Delta	0.0000
Epsilon	0.0000	Epsilon	0.0000	Epsilon	0.0000	Epsilon	0.0000
Zeta	0.0000	Zeta	0.0000	Zeta	0.0000	Zeta	0.0000
Eta	0.0000	Eta	0.0000	Eta	0.0000	Eta	0.0000
Theta	0.0000	Theta	0.0000	Theta	0.0000	Theta	0.0000
Iota	0.0000	Iota	0.0000	Iota	0.0000	Iota	0.0000
Kappa	0.0000	Kappa	0.0000	Kappa	0.0000	Kappa	0.0000
Lambda	0.0000	Lambda	0.0000	Lambda	0.0000	Lambda	0.0000
Mu	0.0000	Mu	0.0000	Mu	0.0000	Mu	0.0000
Nu	0.0000	Nu	0.0000	Nu	0.0000	Nu	0.0000
Xi	0.0000	Xi	0.0000	Xi	0.0000	Xi	0.0000
Omicron	0.0000	Omicron	0.0000	Omicron	0.0000	Omicron	0.0000
Pi	0.0000	Pi	0.0000	Pi	0.0000	Pi	0.0000
Rho	0.0000	Rho	0.0000	Rho	0.0000	Rho	0.0000
Sigma	0.0000	Sigma	0.0000	Sigma	0.0000	Sigma	0.0000
Tau	0.0000	Tau	0.0000	Tau	0.0000	Tau	0.0000
Upsilon	0.0000	Upsilon	0.0000	Upsilon	0.0000	Upsilon	0.0000
Phi	0.0000	Phi	0.0000	Phi	0.0000	Phi	0.0000
Chi	0.0000	Chi	0.0000	Chi	0.0000	Chi	0.0000
Psi	0.0000	Psi	0.0000	Psi	0.0000	Psi	0.0000
Omega	0.0000	Omega	0.0000	Omega	0.0000	Omega	0.0000

VERIFIED STATEMENT (DECLARATION) CLAIMING SMALL-ENTITY STATUS

I, the undersigned, do hereby declare that:

☒ I am an independent inventor as defined in 37 CFR 1.9(c) for purposes of paying reduced fees to the Patent and Trademark Office with regard to the invention described in the patent or application identified above.

The individual, concern or organization identified above has not assigned, granted, conveyed or licensed, and is under no obligation under contract or law to assign, grant, convey or license, any rights in the invention to any person who would not qualify as an independent inventor under 37 CFR 1.9(c) if that person had made the invention, or to any concern which would not qualify as a small business concern under 37 CFR 1.9(d) or a nonprofit organization under 37 CFR 1.9(e).

If the rights held by the above-identified individual, concern or organization are not exclusive, each individual, concern or organization having rights in the invention are identified below. Each such individual, concern or organization must file separate verified statements averring to their status as small entities.

***NOTE: Separate verified statements are required from each named person, concern or organization having rights to the invention averring to their status as small entities. (37 CFR 1.27).**

FULL NAME: Brad Silen

ADDRESS: #5 South Forty Dock, Sausalito, CA 94965

☒ INDIVIDUAL ☐ SMALL BUSINESS CONCERN ☐ NONPROFIT ORGANIZATION

FULL NAME: James Allen Group

ADDRESS: 1335 Cristina Avenue, San Jose, California 95125

☐ INDIVIDUAL ☒ SMALL BUSINESS CONCERN ☐ NONPROFIT ORGANIZATION

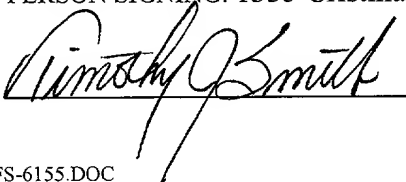
I acknowledge the duty to file, in this application or patent, notification of any change in status resulting in loss of entitlement to small-entity status prior to paying, or at the time of paying, the earliest of the issue fee or any maintenance fee due after the date on which status as a small entity is no longer appropriate. (37 CFR 1.28(b)).

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under section 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of the application, any patent issuing thereon, or any patent to which this verified statement is directed.

NAME OF PERSON SIGNING: Timothy J. Smith

TITLE OF PERSON (if not an owner or individual):

ADDRESS OF PERSON SIGNING: 1335 Cristina Avenue, San Jose, California 95125

SIGNATURE: 

DATE: OCTOBER 27, 2000

HAYSCH.002A

SPECIFICATION FOR

METHODS AND APPARATUSES FOR SYNCHRONIZING MIXED-MEDIA DATA FILES

5

BACKGROUND

1. Field of the Invention.

The invention relates generally to the field of media presentation, and more particularly to methods and apparatus for synchronizing media files.

2. Background Information.

A mixed-media file presentation system is a presentation system that simultaneously presents different media elements, such as video, audio, images, and text. Figure 1 illustrates such a mixed-media file presentation system as the display screen of a computer, which is one means for delivering a mixed-media file presentation. The mixed-media file presentation system of Figure 1 comprises a frame and background 106 that hold a variety of media elements. In this particular example, there is a video element 100, a static image element 102, and a text element 104 in the form of subtitles. When the mixed-media file presentation system is in use, the video plays while the slides and subtitles change. The system may also present animations or simulations.

In order for the mixed-media presentation to flow smoothly, it is necessary to synchronize the different media elements so that they will be presented at the correct moment in the presentation. For instance, the presentation can consist of a video of a person giving a lecture, slides to which the lecturer refers, and subtitles that display what the lecturer is saying in any of a number of languages.

The presentation system must display the slides at the correct moment in the lecture, and it must correctly present subtitles while the video is playing. The presentation must correctly associate these media elements to one another.

According to the present state-of-the-art, methods exist for achieving this type of synchronization. In a known embodiment, data is physically embedded for the slides and subtitles into the actual video data file. For example, a user will embed Universal Record Locators ("URLs") at locations within the video data file that are links to the slides or subtitles, or the user will place URLs at the front of the video data file and embed markers into the video data file at various locations in the data stream. The embedded URLs or markers appear at locations in the video data file that correspond to when the slides and subtitles are to appear.

A drawback to the above approach is that it is destructive and therefore makes reuse of the files a problem. By stating

that this method is destructive, it is meant that it permanently changes the video data file (for instance). Later, if it is desired to reuse some portion of the previous files, then the user must start from scratch and embed new data into a fresh
5 video data file.

Notwithstanding the destructive nature of such a system, the method for making changes to a mixed-media file presentation system discussed above is time-consuming and tedious: users often must carefully watch and time a video as it plays to know where to embed the slide and subtitle data.

SUMMARY OF THE INVENTION

The invention is a method and apparatus for synchronizing media files in a non-destructive manner. According to an
15 aspect, the invention enables changes to be made to a mixed-media presentation without creating a new video data file.

According to one embodiment, a method for synchronizing mixed-media data files comprises receiving at least one streaming media file, receiving at least one static media file,
20 processing the streaming media file to produce a streaming output for a user, querying the streaming media file to acquire a time marker upon receiving a user input, and generating an output file that links the static media file with the time marker and the streaming media file. The time marker represents

a point where the static media file is to be synchronized to the streaming media file. The output file can be used by a mixed-media file presentation system to synchronize the static media file with the streaming media file, or it can be used during the building of a mixed-media file presentation system.

In another embodiment, the invention is a computer system that includes a database comprising one or more data structures relating to streaming media files and static media files, and an application program coupled to the database that is configured to execute the method described above.

BRIEF DESCRIPTION OF THE FIGURES

Figure 1 is a mixed-media file presentation system as displayed on a monitoring device.

Figure 2 illustrates a network architecture in which the invention can be deployed.

Figure 3 is a flowchart depicting overall steps for synchronizing media files according to an embodiment of the invention.

Figure 4 illustrates an embodiment of a user interface for the sync engine.

Figure 5A is a flowchart depicting steps for synchronizing static media files to a streaming media file according to an embodiment of the invention.

Figure 5B is a flowchart depicting steps for synchronizing transcript files to a streaming media file and creating subtitles according to an alternate embodiment of the invention.

Figure 6A depicts a content definition file containing
5 information related to a static media file.

Figure 6B depicts a content definition file containing information related to a transcript file.

Figure 7 depicts the relationship between the static media files and the transcript files as synchronized to the streaming
10 media file.

Figure 8 is a flowchart depicting an alternate method for synchronizing media files.

Figure 9 is a graphical representation of the alternate method of synchronizing media files shown in Figure 8.

Figure 10 is a flowchart illustrating how a mixed-media
15 file presentation system is generated.

Figure 11 illustrates how a database is built and modified in accordance with the overall process depicted in Figure 10.

Figure 12 illustrates an embodiment of a computer system
20 configured to employ the invention.

DETAILED DESCRIPTION OF THE INVENTION

The invention is directed to methods and apparatuses for synchronizing media files to enable them to be concurrently

presented in a mixed-media file presentation system. The apparatus of the invention, as well as the associated methods, utilize non-destructive techniques to synchronize media files, wherein static media files and transcript files synchronization to a streaming media file occurs without any destructive changes being made to the streaming media file itself. The invention allows for the synchronization of any types of static media files and transcript files, such as slides, subtitles, and other types of images or animations, with any type of streaming media file, such as video or audio.

1. Terminology

The term "media file" refers to data files that convey sensory information to a person, such as visual information or audible information. The visual information can take the form of videos, animations, simulations, graphics, images, and text. The audible information can take the form of any type of audio information, including human speech, sound effects.

The term "streaming media file" refers to media that is continuous in nature and that has a time component to it, such media primarily being video files and audio files. Streaming media files can be queried as they run to determine how much time has elapsed since they were started. Representative streaming media files include RealVideo files, RealAudio files,

Quicktime files, MPEG files, Windows Media files, MP3 files, Wave files, and any other computer generated video and audio files.

The term "static media file" refers to media that is static in nature. Static media files include graphics files such as GIF images, JPEG images, Bitmap files, or Computer-Aided Design ("CAD") drawings, as well as animation files that do not have a time component, such as GIF animations or Flash animations. Static media files also include text files, such as Word documents, WordPerfect documents, Portable Document Files ("PDF"), and other computer generated text documents.

The term "mixed-media" refers to a combination of both streaming media and static media files.

The term "transcript file" refers to a static media text
file used as a speech transcript during a streaming
presentation. Transcript files can be in any of the text
formats described herein, and can also be in a standard
generalized markup language ("SGML"), such as hypertext markup
language ("HTML") format. Transcript files are a subset of
static media files, but are herein generally referred to
separately because the invention often handles transcript files
differently than other static media file types. For example,
the methods and apparatus of the invention use the transcript
files to create subtitles for mixed-media presentations.

The term "time marker" refers to a numerical or other symbolic value obtained from a streaming media file when the streaming media file is queried. The numerical value generally represents a quantity or unit of time that has elapsed as measured from the starting point of the streaming media file to the point where the streaming media file was queried, or since the last time marker.

The term "server" refers to software on a computer system or device on a network that receives and answers requests for information. Typically, the computer system is also dedicated to storing data files and managing network resources, including network traffic. The computer system or device generally includes a processor and a data storage device, and is typically connected to the Internet.

2. Network Architecture Overview

Figure 2 depicts a network architecture in which the invention can be employed. Generally, a user creates media files on a computer system 200 and transfers them to a synchronization server 216 that carries out the methods described herein. The media files include streaming media files 202, static media files 204, and transcript files 206. A communications network, such as the Internet 212, carries out the transmitting of media files using communication lines 222.

These communication lines 222 can comprise phone lines, ISDN lines, cable lines, T-1 lines, T-3 lines, E-1 lines, or any other communication lines used to transfer data. Other communications networks, including local area networks (e.g. company intranets) or point-to-point connections can be used as well.

To move the media files to synchronization server 216, it is preferred that the user first upload static media files 204, transcript files 206, and streaming media files 202 to a server 208. Server 208 has a link to a communications network, such as the Internet 212, via communication lines 222. By placing the media files on server 208, synchronization server 216 can access the media files whenever necessary to carry out the methods without having to exhaust it's own data storage space to store these media files. In addition, the use of server 208 is preferred because if the media files are resident on the user computer system 200, synchronization server 216 only has access to the media files when the user computer system 200 is powered up and connected to the Internet. Server 208 can be owned and operated by an ISP, or can belong to the user.

Synchronization server 216 carries out the methods of the invention using streaming media files 202, static media files 204, and transcript files 206 created by the user.

Synchronization server 216 comprises a computer system, such as computer system 200.

Data storage device 220 in the synchronization server 216 preferably stores a sync engine 224. Sync engine 224 is a computer program comprising one or more sequences of instructions that cause a processor to perform a number of acts in accordance with the methods described herein. It is also preferred that the data storage device 220 stores a database 226, which is discussed in further detail below.

There are several alternate embodiments of the invention. For instance, the streaming media files 202 can be stored on a streaming server, on synchronization server 216, or on a user computer system 200, instead of on server 208. A streaming server is a server that uses specially-designed hardware and/or software to quickly and efficiently move streaming media files, which generally have large file sizes.

In other embodiments, the static media files 204 can be stored on the synchronization server 216 or a user computer system 200, instead of on server 208. Generally, the issues that will govern where streaming media files 202 and static media files 204 are stored concern available storage space at the various servers or systems, availability of the files when stored at the different locations, and available bandwidth.

In another embodiment, the computer system 200 can entirely carry out the methods of the invention. In this embodiment, sync engine 224 and database 226 reside on the user computer system 200, and the use of servers, communication lines, and the Internet is avoided.

3. Method for Synchronizing Mixed-Media Files

Figure 3 is a flowchart detailing a conceptual overview of synchronizing media files. Sync engine 224 first receives streaming media files 202, static media files 204, and transcript files 206 (box 300). The transcript files 206 are used by the sync engine 224 to create subtitles for the mixed-media presentation.

Next, as shown in box 302, the sync engine 224 receives a system input that selects which streaming media file 202 to use for the synchronization process. Moving to box 304, the sync engine 224 then receives a system input that selects which transcript language to use when creating subtitles for the mixed-media presentation.

Next, as seen in box 306, the sync engine 224 retrieves static media files 204 and the appropriate transcript files 206 associated with the selected streaming media file 202. This act can also depend on the transcript language chosen. Sync engine 224 includes information regarding which static media files 204

and which transcript files 206 are associated with each streaming media file 202. The information is based on relationships specified prior to use of sync engine 224. The process of creating relationships between media files is further
5 described below with reference to Figures 10 and 11.

Turning to box 308, the synchronization process is started. The sync engine 224 concurrently plays streaming media file 202 and displays one or more static media files 204. Alternatively, or in combination with the above playback, one or more portions of transcript files 206 can be played. {Technical question: is
10 the sync engine playing, or is it linking and serving the files it identifies, while a client (e.g., X-server or other) does the actual "playing"?} Next, as shown in box 310, as the streaming media file 202 is playing, the sync engine 224 receives system
15 inputs that establish synchronization points. As used herein, the term "synchronization point" refers to a specific point in time during the presentation of a streaming media file 202 at which a particular static media file 204 or subtitle is made to appear. Thus in a completed mixed-media presentation, for
20 example the mixed-media presentation of Figure 1, as a streaming media file (e.g. video) plays in section 100, each static media file (e.g. slides) or subtitle appears within sections 102 or 104 when its respective synchronization point has been reached.

By the end of the presentation, all of the slides or subtitles will have cycled through and appeared in sections 102 or 104.

Turning to box 312, the sync engine 224 enables the user to pause, rewind, and replay the streaming media file 202. In this manner, the user is able to see where the static media files 204 and subtitles were synchronized to the streaming media file 202. Furthermore, the user is enabled to modify where the media files were synchronized. Moreover, these characteristics allow the user to see if the subtitles were created logically.

Sync engine 224 next generates an output file that contains synchronization data, as shown in box 314. This output file is referred to herein as a "content definition file". A computer program can use the content definition file to build a mixed-media file presentation system, or a mixed-media file presentation system can directly use the content definition file to synchronize media files. The content definition file is described in greater detail with reference to Figures 6A and 6B.

Figure 4 illustrates an embodiment of a user interface 400 that provides a means for a user to interact with sync engine 224. User interface 400 includes a section 402 where a streaming media file 202 is displayed, a section 406 where one or more static media files 204 are displayed, and a section 410 where one or more transcript files 206 are displayed. Within user interface 400, the transcript files 206 appear in a markup

format, shown by each underlined word 412. User interface 400 also includes control buttons 414 for controlling whether streaming media file 202 is playing, paused, or restarted.

According to an embodiment, user interface 400 is designed to run within a web browser, such as Netscape Navigator (obtainable at www.netscape.com) or Microsoft Internet Explorer (obtainable at www.microsoft.com). Such a user interface 400 allows the sync engine 224 to be accessed from remote locations and run over a communications network. Thus, a user on computer system 200 can use the sync engine 224, located on the synchronization server 216, from virtually any location in the world.

Turning to Figure 5A, a flowchart is shown that describes an embodiment of the method for synchronizing a static media file to a streaming media file. Starting with box 500, the sync engine 224 first receives streaming media file 202 and static media file 204. The next steps, shown in boxes 502 and 504, concurrently present the streaming media file 202 and the static media file 204 on a display device. The streaming media file 202 is handled by a processor 204 that, in turn, produces a streaming output. The streaming output is the actual video that is displayed on a display screen or the actual audio that is delivered through a speaker. Thus, a user can see a streaming output such as a video playing in section 402 of the user

interface in Figure 4, while static media file 204 appears in section 406 of the interface 400.

Next, the user establishes a synchronization point to which the static media file 204 is matched. Sync engine 224 is the tool used for this process, and the user interacts with sync engine 224 via user interface 400. Referring to Figure 4 in conjunction with box 506 of Figure 5A, as the streaming media file 202 plays within section 402 of interface 400, the user is allowed to establish a synchronization point by "selecting" static media file 204 at the precise moment the user wants the file 204 to appear during the presentation. This "selection", which is simply an input by the user, can be accomplished in a number of ways. A preferred way is using a mouse to place a pointer over static media file 204 and having the user "click" on static media file 204 at the moment the synchronization point is to be established. This same process can be carried out using a keyboard. The sync engine 224 then receives this input from the user, as shown in box 506.

This process is best illustrated by an example. Consider that a user wants the first slide 204 shown in Figure 4 (the "Customers to Clients" slide) to appear thirty seconds into streaming video 202. The user starts the video and uses a mouse to place a pointer over the "Customers to Clients" slide 204. After thirty seconds of the video has elapsed, the user "clicks"

on the "Customers to Clients" slide 204, thereby establishing a
synchronization point. Now when the completed mixed-media
presentation is shown, the "Customers to Clients" slide appears
on-screen thirty seconds after the video has begun. The user
5 can synchronize all of the slides in this manner, thus making
all of the slides appear at appropriate times during the video.
It is worth noting that the invention can synchronize any type
of static media file to any type of streaming media file in this
manner.

10 Moving on to box 508 of Figure 5A, when an input is
received sync engine 224 queries streaming media file 202 to
obtain a time marker. Upon receiving this query, streaming
media file 202 generates a time marker and sends it to sync
engine 224. {Please explain in technical terms how the time
15 marker is generated. We were told that the sync engine queries
the video for a time marker, but my guess is that the video file
cannot generate a time marker itself, and that another device
performs this process. Please explain what is being queried.}
The time marker indicates how much "run time" has elapsed since
20 the start of the streaming media file. As used herein, the term
run time refers to a quantity of time that has elapsed since the
start of the streaming media file, taking into account only the
actual playing of the streaming media and not including time
incurred due to pauses, rewinds, and fast forwards.

Next, the sync engine receives the time marker (box 510) and generates a content definition file (box 512). The content definition file contains the time marker along with information about the static media file 204 and the streaming media file 202. The content definition file is further described below with reference to Figure 6.

Turning now to Figure 5B, a flowchart is provided that describes a method for creating subtitles from transcript files 206 and synchronizing those subtitles to a streaming media file.

Starting with box 520, the first step in this process is for the sync engine 224 to receive streaming media file 202 and transcript file 206. Turning to box 522, the next step is to convert the individual words of the transcript files into a markup format, such as an HTML format.

The process of converting the transcript files 206 into markup words 412 can be accomplished in a number of ways. A preferred method is to use a computer program, such as a Perl script, to extract each word from the transcript files 206 and re-write each word in an HTML format using anchor tags. This markup process can also be performed using different markup languages.

Moving to boxes 524 and 526 of Figure 5B, the next step in synchronizing the transcript files 206 and creating subtitles is to present streaming media file 202 as a streaming output while

concurrently displaying portions of the one or more transcript files on a display device. Here a user can see the streaming output, such as a video, playing in section 402 of user interface 400, while markup words 412 of the transcript files 206 appear in section 410 of interface 400.

Having the words 412 in markup format allows a user to "select" individual words to synchronize to streaming media file 202. Moreover, it allows the user to designate breaks in the transcript that correspond to the starting and ending points for the subtitles.

Turning to box 528, the sync engine 224 receives inputs that establish synchronization points to which the transcript files 206 are matched and designate the start and end of subtitles. Reviewing Figure 4 in conjunction with box 528 of Figure 5B, as streaming media file 202 plays within section 402 of interface 400, the user is allowed to "select" a markup word 412 of the transcript file 206 at the precise moment the user wants that word to appear as the start of the next subtitle.

An example is now provided to illustrate this process.

Using the markup words 412 in Figure 4, suppose a user wants the following subtitles to be presented at the beginning of the mixed-media presentation:

1. "After you have defined the desired outcome, you can move forward to aligning the problem with a solution."

2. "This is the second step in Designing and Selling a Solution."

3. "and it makes use of a technique that is quite straightforward."

5 To create and synchronize subtitles, the user "selects" the first word of each of the subtitles at appropriate times during the playing of streaming media file 202. So as the lecturer in the streaming video 202 speaks the word "After", the user "selects" markup word "After" 412A. This action establishes a synchronization point for the word "After" 412A, and designates the word "After" 412A as the start of the first subtitle.

10 Next, as the lecturer speaks the word "This", the user "selects" markup word "This" 412B, again establishing a synchronization point for the word "This" 412B and designating the start of the second subtitle. All of the words between "After" 412A and "This" 412B appear in the first subtitle.

15 Finally, when the lecturer speaks the word "and", the user "selects" markup word "and" 412C to again establish a synchronization point and designate the start of the third subtitle. As before, the words between "This" 412B and "and" 412C appear in the second subtitle. The words after "and" 412C appear in the third subtitle. Now when the completed mixed-media presentation is shown, each of the subtitles appear on-

screen at the moment that the lecturer in the video speaks the first word of each subtitle.

Moving on to box 530 of Figure 5B, when sync engine 224 receives an input "selecting" a markup word 412, sync engine 224 creates a break in the markup transcript file 206 to signify the starting point of a subtitle. At approximately the same time, as shown in box 532, sync engine 224 obtains a time marker. The sync engine 224 receives this time marker (box 534) and places it into a content definition file with information describing the transcript file 206 and the streaming media file 202 (box 528).

Figures 6A and 6B are graphical representations of content definition files. Content definition file 600 of Figure 6A is for a static media file 204. This content definition file 600 includes information 602 related to the static media file 204 for which the time marker was generated (e.g. access path information), information 604 relating to streaming media file 202 from which the time marker was obtained (e.g. access path information), and the time marker data 606 itself. Content definition file 600 therefore associates static media file 204 to streaming media file 202, and includes a time marker that designates when static media file 204 is supposed to appear while streaming media file 202 is served. Typically one content

definition file includes data for a plurality of static media files.

Content definition file 620 of Figure 6B includes information 622 relating to the transcript file 206 for which the time markers were generated, information 624 relating to streaming media file 202 from which the time marker was obtained, and time marker data 626 from sync engine 224 that was generated by streaming media file 202. Content definition file 620 also includes information 626 related to breaks in the transcript files 206 that designate the starting and ending points for the subtitles that were created. Similar to content definition file 600 of Figure 6A, content definition file 620 associates the transcript file to streaming media file 202, and includes time markers that designate when each subtitle is to appear while streaming media file 202 is served.

Content definition files 600 and 620 can be used during the building of a mixed-media presentation. For example, a computer program designed to create mixed-media presentations can utilize the information provided by content definition files 600 and 620 during the generation of the mixed-media presentation. One such computer program that can use content definition files 600 and 620 is disclosed in a co-pending U.S. Patent Application Serial No. (not yet assigned), Lyon & Lyon Docket No. 255/194, entitled, "METHODS AND APPARATUS FOR BUILDING A MIXED-MEDIA FILE

PRESENTATION SYSTEM," which is incorporated herein by reference in its entirety. Content definition files 600 and 620 can also be dynamically used by a mixed-media presentation while such a presentation is being served back on a display device.

5 Turning to Figure 7, a graphical representation of how the invention synchronizes a streaming media file 202, several static media files 204, and several subtitles 704 of a transcript file 206 is shown. Static media files 204 and subtitles 704 are synchronized to several different
10 synchronization points 706 along streaming media file 202. So as streaming media file 202 plays and time 808 elapses, each synchronization point 706 is triggered causing a static media file 204 or a subtitle 704 to appear in the mixed-media presentation.

15 According to one embodiment, sync engine 224 resides on a portable storage medium such as a floppy disk or a CD-ROM. This allows sync engine 224 to be used on computer systems other than synchronization server 216. In another embodiment, sync engine
20 224 can reside on storage device 210 of a computer system 200 of the user. This allows a user to use sync engine 224 at their home or office without the need for a link to a communications network such as the Internet.

 In another embodiment, a JavaScript timer is used to measure the elapsed time as the streaming media file 202 when

5 Alternately, user action can be used to synchronize the static media files 204. For instance, a user can input a request to have the next static media file 204 presented by clicking on a markup "Next Page" button on a computer display screen.

15 transcript file 206, and the time marker data.

20 synchronized in accordance with this embodiment.

Turning to Figure 9, streaming media file 202 comprises a series of frames 906, each frame 906 having a unique address. The unique address generally comprises a block number 910 and a frame number 912. For example, frame number 05 of block number

00 has a frame address 914 that comprises a combination of those numbers, such as "00,05" or "0005". The system can then identify a specific frame 906 by locating the block 910 it is contained in, and then finding the frame 912 with the correct frame number.

Turning to box 800 of Figure 8, the sync engine 224 first receives streaming media file 202 and static media file 204. Next, as shown in boxes 802 and 804, the sync engine 224 plays streaming media file 202 while concurrently displaying static media files 204. Again, the streaming media file 202 is presented as a streaming output after being handled by a processor 204.

Moving to box 806, the sync engine 224 receives an input that establishes a synchronization point. When this input is received, as shown in box 808, sync engine 224 queries streaming media file 202 to obtain a sync frame. As used herein the term "sync frame" refers to a frame 906 of the streaming media file 202 that is in use (i.e. being served) at the moment the sync engine 224 receives the input. Upon receiving this query, streaming media file 202 identifies a sync frame and sends its unique address to sync engine 224. Then as shown in boxes 810 and 812, sync engine 224 receives the sync frame information and generates a content definition file containing the sync frame's

unique address as well as information about the static media file 204 and streaming media file 202.

This embodiment can also be used to synchronize transcript files to a streaming media file 202 and to create subtitles. In this embodiment, when the user "selects" a markup word 412 to create a synchronization point and to designate the start of the next subtitle, the sync engine 224 obtains a sync frame. As before, a content definition file is created storing this information.

Turning again to Figure 9, static media files 204 and subtitles 704 are synchronized to several sync frames 908 along streaming media file 202. Each sync frame 908 has a unique address 916. As streaming media file 202 plays and frames 906 elapse, each sync frame 908 is played causing a static media file 204 or a subtitle 704 to appear in the mixed-media presentation. It is the unique addresses 916 of the sync frames 908 that are stored in the content definition files by the sync engine 224.

4. Overall Process Flow for Building a Mixed-Media File
Presentation System

Figures 10 and 11 illustrate the process for building a mixed-media file presentation system. Figure 10 is a flowchart describing how a mixed-media file presentation system is generated. Figure 11 illustrates how database 226, within the synchronization server 216, is built and modified as the process is carried out.

Starting at box 1000 of Figure 10, a user begins by defining the content structure of the presentation. The user decides what the content of the mixed-media presentation will be, what types of streaming media and static media need to be created, and how the overall process will flow.

Next, as shown in box 1002, the user creates static media files 204 for use in the mixed-media file presentation system. Typically static media files 202 comprise slides, images, drawings, simulations, or other media files that complement the content of streaming media file 202.

As shown in box 1004, the user then creates one or more transcript files 206 that will be used in the streaming media file of the mixed-media presentation. These can be transcripts of videos that will be presented, or they can be transcripts of an audio-only presentation. The transcripts can then be translated into other languages. These translations create

multiple sets of subtitles for the mixed-media presentation, wherein each set is in a different language. This allows users of the mixed-media presentation who are proficient in languages other than the original language of the presentation to still
5 use the mixed-media presentation.

Turning to box 1006, the user records the streaming video or audio files for use in the mixed-media presentation. Generally a video format implies that there is associated audio, although a user can create a silent video. The transcripts
10 generated in box 1004 are typically used when creating the video or audio files. Next, as shown in box 1008, the user converts the streaming video or audio files into different computer-readable streaming media formats, such as RealVideo, RealAudio, Quicktime, Windows Media, or any other computer-readable format.
15 The creation of several different formats helps to ensure that the final mixed-media file presentation system will be supported by a wide variety of user platforms.

Turning to box 1010, the different streaming media files 202, static media files 202, and transcript files 206 created
20 are sent to a server where they are made available to sync engine 224. The use of a server allows the sync engine 224 to access these files whenever needed.

As shown in box 1012, after creating all of the media files, the next step is to generate relationships between the

media files. A user can accomplish this by placing information about the media files, including the media files themselves, into database 226 and then creating a directory tree structure 1112 within database 226 that relates all of the media files to one another. The process of creating a database and generating relationships among the different media files is further described with reference to Figure 11. Alternately, an authoring tool can be used to create the relationships.

Moving to box 1014, after the relationships between the media files are generated, the next step is to deliver the information in database 226 to sync engine 224. The sync engine 224 then obtains time markers that synchronize static media files 204 and transcript files 206 to streaming media files 202.

Turning to box 1016 now, content definition files are extruded from database 226 and are delivered to a build engine of a mixed-media file presentation system.

The content definition files can be generated as an output data file from database 226 and provide information to the build engine that the build engine uses to construct a mixed-media file presentation system. The content definition files are extruded from database 226 in a format that the build engine can understand, for example Extensible Mark-Up Language (XML).

As shown in box 1018, the build engine then uses the content definition files, as well as stylesheets generated in

box 1022, to generate a mixed-media file presentation system.
The build engine can deploy the mixed-media file presentation
system over a communications network such as an intranet or the
Internet, or to a portable medium such as a CD-ROM. This is
5 shown in box 1020.

Methods and apparatus for using a build engine to generate
a mixed-media file presentation system are more fully described
in a co-pending U.S. Patent Application Serial No. (not yet
assigned), Lyon & Lyon Docket No. 255/194, entitled "METHODS AND
10 APPARATUS FOR BUILDING A MIXED-MEDIA FILE PRESENTATION SYSTEM",
which is incorporated herein by reference above.

Turning to Figure 11, it is a graphical representation of
how information is gathered and how relationships between the
media files can be created.

15 Starting with step (a), the transcript files 206 that were
created are stored in database 226. This can include the
original transcript files as well as any translations.
Alternately, in lieu of having database 226 store transcript
files 206, database 226 can simply store information regarding
20 the location or access paths of the transcript files 206, and
the transcripts files 206 themselves can be stored elsewhere,
such as on server 208 or on the user computer system 200. This
alternative conserves storage space at synchronization server
216.

Next, moving to step (b), the location information of streaming media files 202 is stored in database 226. The location information is preferably in the form of access paths 202' that identify the location of streaming media files 202 on server 208. Thus, database 226 now comprises transcript files 206 as well as the access paths 202' to streaming media files 202 stored on server 208. Alternately, streaming media files 202 can be stored in database 226, rather than being stored to server 208.

Staying at step (b), as is the case with streaming media files 202, access paths 204' to static media files 204 are stored in database 226. Database 226 now comprises transcript files 206, access paths 202' to streaming media files 202, and access paths 204' to static media files 204. In alternate embodiments, static media files 204 can reside in database 226 rather than being deployed to server 208, in lieu of access paths 204' being stored in database 226.

Moving on to step (c), relationships are created that associate static media files 204 and transcript files 206 to streaming media files 202. It is these relationships that allow sync engine 224 to pull up the appropriate static media files 204 and transcript files 206 when a streaming media file 202 is chosen.

For instance, the user can first create an overall tree structure 1112 for the mixed-media presentation. As shown by the example in step (c), the overall tree structure 1112 can comprise one section 1108 with three different lessons 1110.

5 The user can then determine which media files 206, 202, and 204 should be placed into each lesson 1110. In this example, one streaming media file 202 is being associated with each lesson 1110. Next, the corresponding transcript file 206 and a plurality of static media files 204 are associated with each lesson 1110. Generally, each of the transcript files 206 and each of the static media files 204 were originally created to be associated with a particular streaming media file 202. It is through the relationship generating process of box 1012 that the user can establish these associations for use by the sync engine 15 224.

Since the overall tree structure 1112 only uses access paths 202' and 204' for the streaming media 202 and static media 204 files, each media file can be used in more than one lesson 1110 or section 1108, and the user can specify this in the overall file structure 1112. This is possible because the actual media files are located on server 208, and since the actual media files are not themselves placed into the overall file structure 1112, the media files are available for use in multiple lessons 1110 or sections 1108.

Moving on to step (d), the time markers 1114 generated by the sync engine 224 are added to database 226 where they are associated with their respective static media files 204 or transcript files 206. The result of this database building process is a directory structure defining which static media files 204 and transcript files 206 are associated with which streaming media files 202, and how the media files are laid out in an overall file structure 1112.

5. Hardware Overview

Figure 12 is a block diagram of an exemplary computer system 1200 upon which methods of the invention can be implemented.

Computer system 1200 includes a bus 1202, or other communication mechanism for communicating information, and a processor 1204 coupled with bus 1202 for processing information. Computer system 1200 also includes a main memory 1206, such as a random access memory ("RAM"), or other dynamic (or "volatile") storage device, coupled to bus 1202. The main memory 1206 stores information and instructions executed by processor 1204 during execution. Main memory 1206 also stores temporary variables or other intermediate information during execution of instructions by processor 1206.

Computer system 1200 further includes a read only memory ("ROM") 1208 or other static (or "persistent") storage device

09699600
40
5 (e.g., FLASH, PROM, EEPROM, etc.) coupled to bus 1202. The ROM 1208 stores static information and instructions for processor 1204, in particular the browser as described herein. It is worth noting that one or more banks of memory can comprise ROM 1208. A storage device 1210 (or "hard disk", or "hard drive"), such as a magnetic disk or optical disk, is coupled to bus 1202. The storage device 1210 stores information such as data structures and instructions, for example data structures for streaming media files 202 and static media files 204, or structures relating to the operating system or application programs that use the operating system.

15 Computer system 1200 is preferably coupled via bus 1202 to a display device 1212, such as a cathode ray tube ("CRT") or an active or passive-matrix display. The display 1212 presents images to an end-user. An input device 1214, including alphanumeric and other keys, is coupled to bus 1202. The input device 1214 communicates information and command selections to processor 1204. Another type of user input device is cursor control 1216, such as a mouse, a trackball, or cursor direction
20 keys, for communicating direction information and command selections to processor 1204 and for controlling cursor movement on display 1212. This input device 1214 typically has two degrees of freedom in two axes, a first axis (e.g., x) and a second axis (e.g., y), that allows the device to specify positions in a plane.

According to an aspect of the invention, the processor 1204 in the computer system 1200 executes one or more sequences of instructions contained in main memory 1206. Such instructions are read into main memory 1206 from another computer-readable medium, such as storage device 1210 or ROM 1208. The instructions can be executable object code or interpreted code that is processed by a run-time engine (e.g., Javascript).

Execution of the sequences of instructions contained in main memory 1206 causes processor 1204 to perform the methods of the invention as described herein. In alternative embodiments, hard-wired circuitry can be used in place of or in combination with software instructions to implement the invention. Thus, embodiments of the invention are not limited to any specific combination of hardware circuitry and software.

The term "computer-readable medium" as used herein refers to any medium that participates in providing instructions to processor 1204 for execution. Such a medium can take many forms, including but not limited to, non-volatile media, volatile media, and transmission media. Non-volatile media includes, for example, optical or magnetic disks, such as storage device 1210. Volatile media includes dynamic memory, such as main memory 1206.

Transmission media includes coaxial cables, copper wire and fiber optics, including the wires that comprise bus 1202.

Transmission media can also take the form of acoustic or light

waves, such as those generated during radio-wave and infrared data communications.

Common forms of computer-readable media include, a floppy disk, a flexible disk, a hard disk, a magnetic tape, or any other magnetic media, a CD-ROM, any other optical media, punchcards, a paper-tape, any other physical media with patterns of holes, a RAM, a ROM, a FLASH, or any other memory chip or cartridge, a carrier wave as described hereinafter, or any other media from which a computer can read.

Various forms of computer-readable media can be involved in carrying one or more sequences of one or more instructions to processor 1204 for execution. For example, the instructions can initially be carried on a magnetic disk of a remote computer. The remote computer can load the instructions into its dynamic memory and send the instructions over a telephone line using a modem. A modem local to computer system 1200 can receive the data on the telephone line and use an infrared transmitter to convert the data to an infrared signal. An infrared detector coupled to bus 1202 can receive the data carried in the infrared signal and place the data on bus 1202. Bus 1202 carries the data to main memory 1206, from which processor 1204 retrieves and executes the instructions. The instructions received by main memory 1206 can optionally be stored on storage device 1210 before or after execution by processor 1204.

Computer system 1200 also includes a communication interface 1218 coupled to bus 1202. Communication interface 1218 provides a two-way data communication coupling to a network link 1220 that is connected to a local network 1222. For example, communication interface 1218 can be an integrated services digital network ("ISDN") card or a modem to provide a data communication connection to a corresponding type of telephone line. As another example, communication interface 1218 can be a local area network ("LAN") card to provide a data communication connection to a compatible LAN. Wireless links can also be implemented. In any such implementation, communication interface 1218 sends and receives electrical, electromagnetic or optical signals that carry digital data streams representing various types of information.

Network link 1220 preferably provides data communication through one or more networks to other data devices. For example, network link 1220 can provide a connection through local network 1222 to a host computer 1224 or to data equipment operated by an Internet Service Provider ("ISP") 1226. ISP 1226 in turn provides data communication services through the "Internet" 1228 -- for example computer diagnostic services. Local network 1222 and Internet 1228 both use electrical, electromagnetic or optical signals that carry digital data streams. The signals through the various networks and the signals on network link 1220 and through communication interface

1218, which carry the digital data to and from computer system 1200, are exemplary forms of carrier waves transporting the information.

Computer system 1200 can send messages and receive data, including program code, through the network(s), network link 1220 and communication interface 1218. In the Internet example, a server 1230 might transmit requested code for an application program through Internet 1228, ISP 1226, local network 1222 and communication interface 1218 -- for example using the FTP protocol. In accordance with the invention, one such downloaded application is executable software code or computer configuration parameters that perform the methods of the invention.

The received code can be executed by processor 1204 as it is received, and/or stored in main memory 1206, storage device 1210, or other non-volatile storage for later execution. In this manner, computer system 1200 can obtain application code in the form of a carrier wave.

Methods and apparatus for synchronizing media files to enable them to be concurrently presented in a mixed-media file presentation system have been disclosed. In the foregoing specification, the invention has been described with reference to specific embodiments thereof. It will, however, be evident that various modifications and changes can be made thereto

CLAIMS

What is claimed is:

1. A method for synchronizing media files, comprising:
receiving a streaming media file;
5 receiving a static media file;
producing a streaming output from the streaming media file;
querying the streaming output for a time marker; and
associating the static media file with the time marker and
the streaming media file in an output file.

2. The method of claim 1, further comprising receiving an
input that designates a point in the streaming output to which
the static media file is to be synchronized.

15 3. The method of claim 1, further comprising generating a
time marker that indicates a quantity of time that has elapsed.

20 4. The method of claim 3, wherein the quantity of time is
measured between a first point in time, relating to when the
streaming output was started, and a second point in time,
relating to when the user input was received.

5. The method of claim 1, further comprising displaying the streaming output synchronized with one or more static media files based upon one or more associations in the output file.

5 6. The method of claim 1, wherein the streaming media file is selected from the group consisting of video data files, and audio data files.

7. The method of claim 1, wherein the streaming output is selected from the group consisting of streaming video and streaming audio.

8. The method of claim 1, wherein the static media file is selected from the group consisting of graphic data files, text data files, and non-streaming animation files.

9. A computer-readable medium having stored therein one or more sequences of instructions for synchronizing media files, the one or more sequences of instructions causing one or more processors to perform a number of acts, said acts comprising:

receiving a streaming media file;

receiving a static media file;

producing a streaming output from the streaming media file;

querying the streaming output for a time marker upon
receiving an input; and

associating the static media file with the time marker and
the streaming media file in an output file.

5

10. The computer readable medium of claim 9, the method
further comprising receiving an input that designates a point in
the streaming output to which the static media file is to be
synchronized.

11. The computer readable medium of claim 9, the method
further comprising generating a time marker that indicates a
quantity of time that has elapsed.

12. The computer readable medium of claim 9, wherein the
quantity of time is measured between a first point in time,
relating to when the streaming output was started, and a second
point in time, relating to when the user input was received.

13. The computer readable medium of claim 9, the method
further comprising displaying the streaming output synchronized
with one or more static media files based upon one or more
associations in the output file.

14. The computer readable medium of claim 9, wherein the streaming media file is selected from the group consisting of video data files and audio data files.

5 15. The computer readable medium of claim 9, wherein the streaming output is selected from the group consisting of streaming video and streaming audio.

10 16. The computer readable medium of claim 9, wherein the static media file is selected from the group consisting of graphic data files, text data files, and non-streaming animation files.

15 17. A method for synchronizing media files, comprising:
receiving a streaming media file that comprises a series of frames, each frame having a unique address;
receiving a static media file;
producing a streaming output from the streaming media file;
querying the streaming output for a sync frame; and
20 associating the static media file with the sync frame and the streaming media file in an output file.

18. The method of claim 17, further comprising receiving an input that designates a point in the streaming output to which the static media file is to be synchronized.

5 19. The method of claim 18, further comprising identifying a sync frame that comprises a frame of the streaming media file corresponding to the point in the streaming output designated by the input.

10 20. The method of claim 17, further comprising displaying the streaming output synchronized with one or more static media files based upon one or more associations in the output file.

15 21. A computer-readable medium having stored therein one or more sequences of instructions for synchronizing media files, the one or more sequences of instructions causing one or more processors to perform a number of acts, said acts comprising:

receiving a streaming media file that comprises a series of frames, each frame having a unique address;

20 receiving a static media file;

producing a streaming output from the streaming media file;

querying the streaming output for a sync frame upon receiving an input; and

associating the static media file with the sync frame and the streaming media file in an output file.

22. The computer readable medium of claim 21, the method
5 further comprising receiving an input that designates a point in the streaming output to which the static media file is to be synchronized.

23. The computer readable medium of claim 21, the method
10 further comprising identifying a sync frame that comprises a frame of the streaming media file corresponding to the point in the streaming output designated by the user input.

24. The computer readable medium of claim 21, wherein the
15 step of associating further comprises associating the static media file with the unique address of the sync frame.

25. A method for synchronizing media files, comprising:
receiving a streaming media file;
20 receiving a transcript data file;
producing a streaming output from the streaming media file;
producing a markup transcript from the transcript data
file;

querying the streaming output for a time marker upon
receiving an input; and

associating a selected portion of the markup transcript
with the time marker and the streaming media file.

5

26. The method of claim 25, further comprising displaying
the streaming output synchronized with one or more portions of
the markup transcript based upon one or more associations in the
output file.

10

27. The method of claim 25, further comprising receiving
an input that selects a portion of the markup transcript and
identifies a point in the streaming output to which the selected
portion of the markup transcript is to be synchronized.

15

28. A method for synchronizing media files, comprising:

receiving a streaming media file that comprises a series of
frames, each frame having a unique address;

receiving a transcript data file;

20 producing a streaming output from the streaming media file;

producing a markup transcript from the transcript data
file;

querying the streaming output for a sync frame upon
receiving an input; and

associating a selected portion of the markup transcript with the sync frame and the streaming media file in an output file.

5 29. The method of claim 28, further comprising receiving an input that selects a portion of the markup transcript and identifies a point in the streaming output to which the selected portion of the markup transcript is to be synchronized.

10 30. The method of claim 28, further comprising displaying one or more portions of the markup transcript.

15 31. A computer system for synchronizing media files, comprising:

a computer that comprises:

a processor;

a main memory communicatively coupled to the processor; and

a storage device communicatively coupled to the

20 processor;

a database running on the computer from the main memory, the database comprising:

one or more data structures relating to one or more streaming media files stored in the storage device; and

one or more data structures relating to one or more static media files stored in the storage device; and an application program coupled to the database and configured to support a user, the application program configured

5 to:

```
produce a streaming output from a streaming media
file;
```

query a streaming media file for a time marker upon receiving an input; and

associate the static media file with the time marker
and the streaming media file in an output file.

ABSTRACT

A method for synchronizing mixed-media data files that comprises receiving a streaming media file, receiving a static media file, producing a streaming output from the streaming media file, querying the streaming output for a time marker upon
5 receiving an input, and associating the static media file with the time marker and the streaming media file in an output file.

An apparatus for synchronizing mixed-media data files that comprises a computer system including a database with one or more data structures relating to streaming media files and static media files stored on a storage device. The computer system also includes an application program coupled to the database that is configured to execute the methods described above.

FIGURE 1

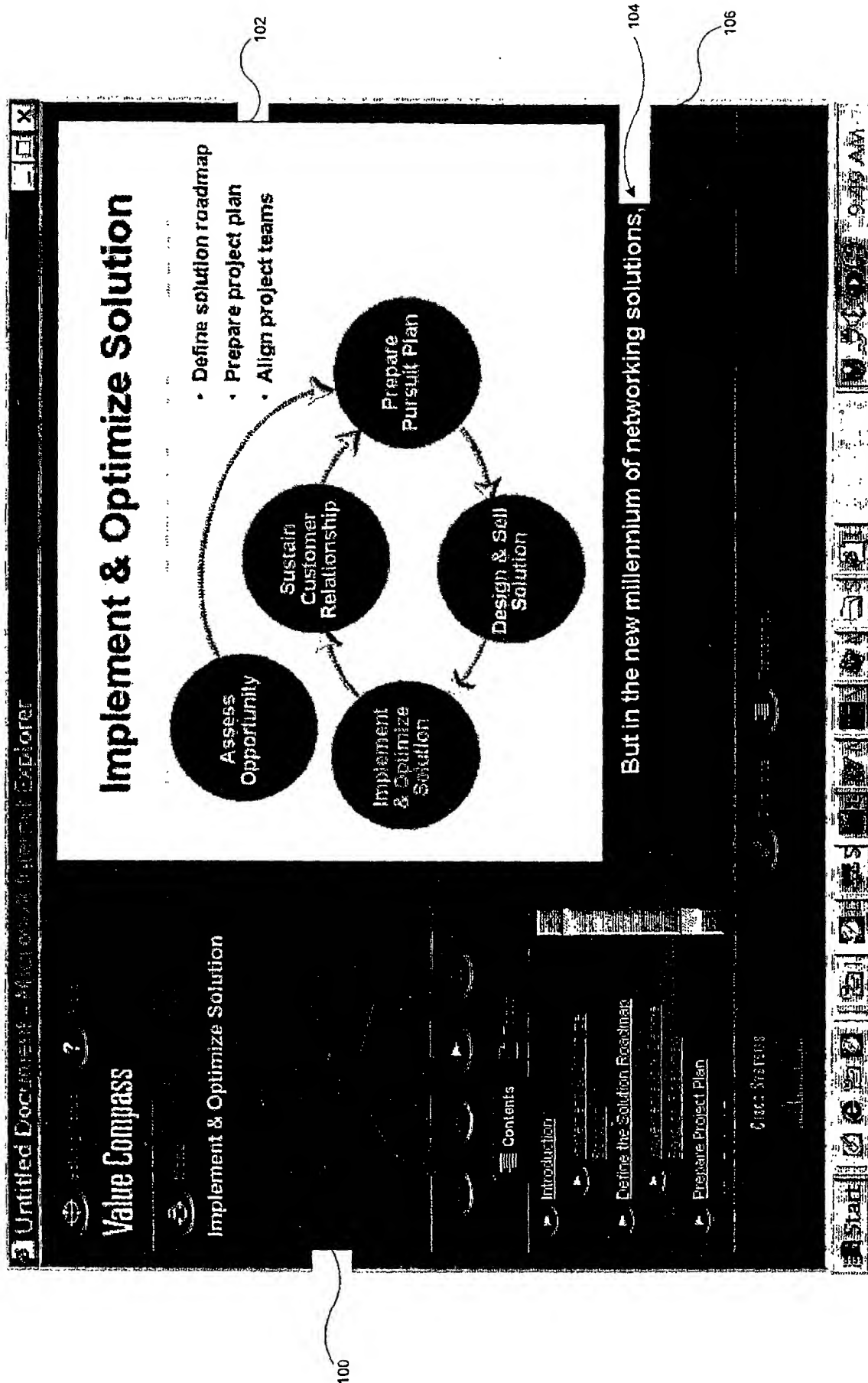


FIGURE 2

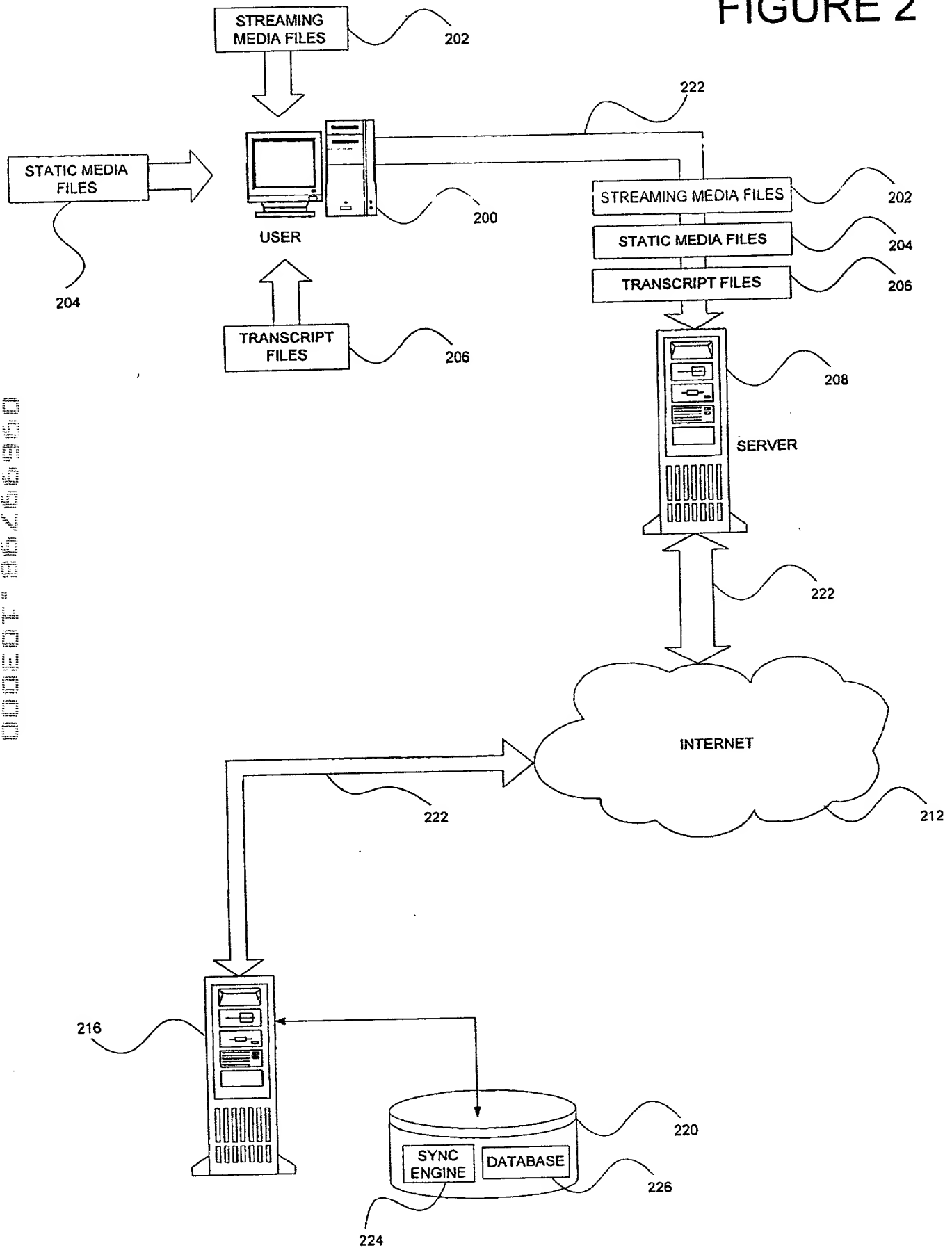
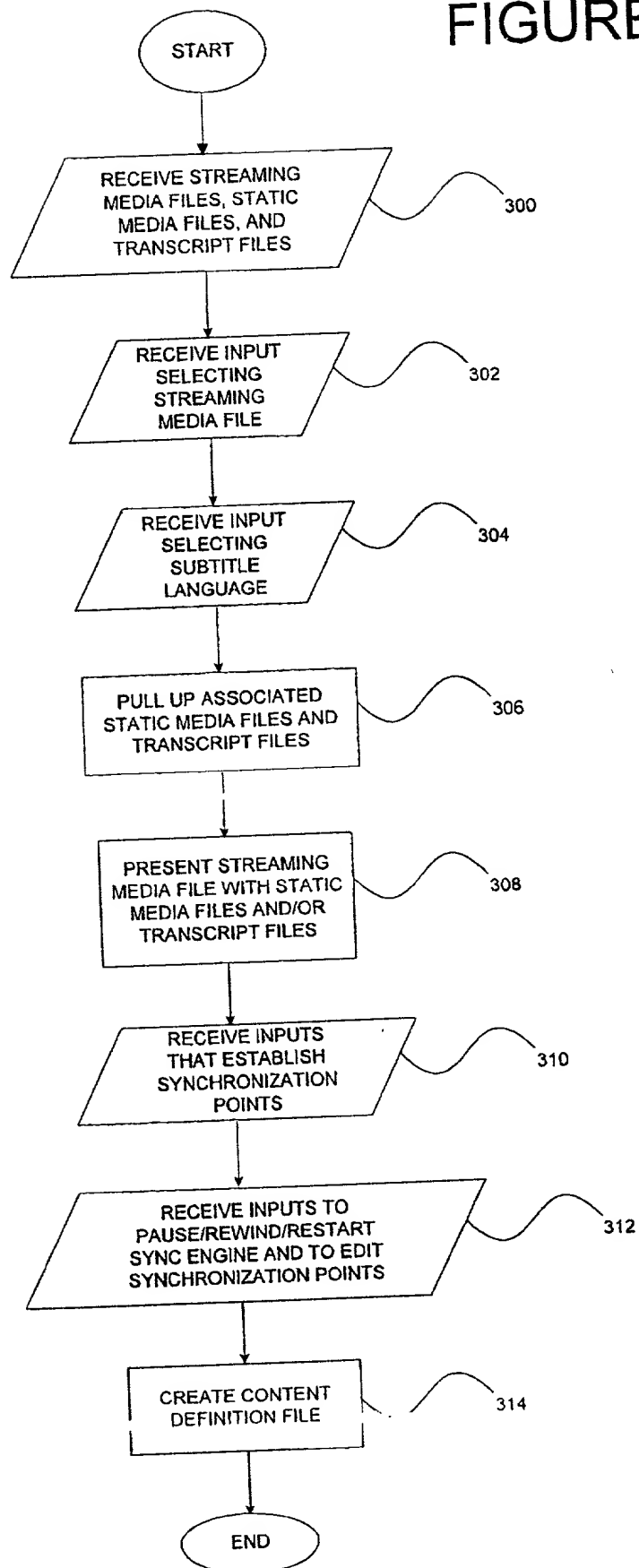


FIGURE 3



CODE OF 1949 FIGURE 4

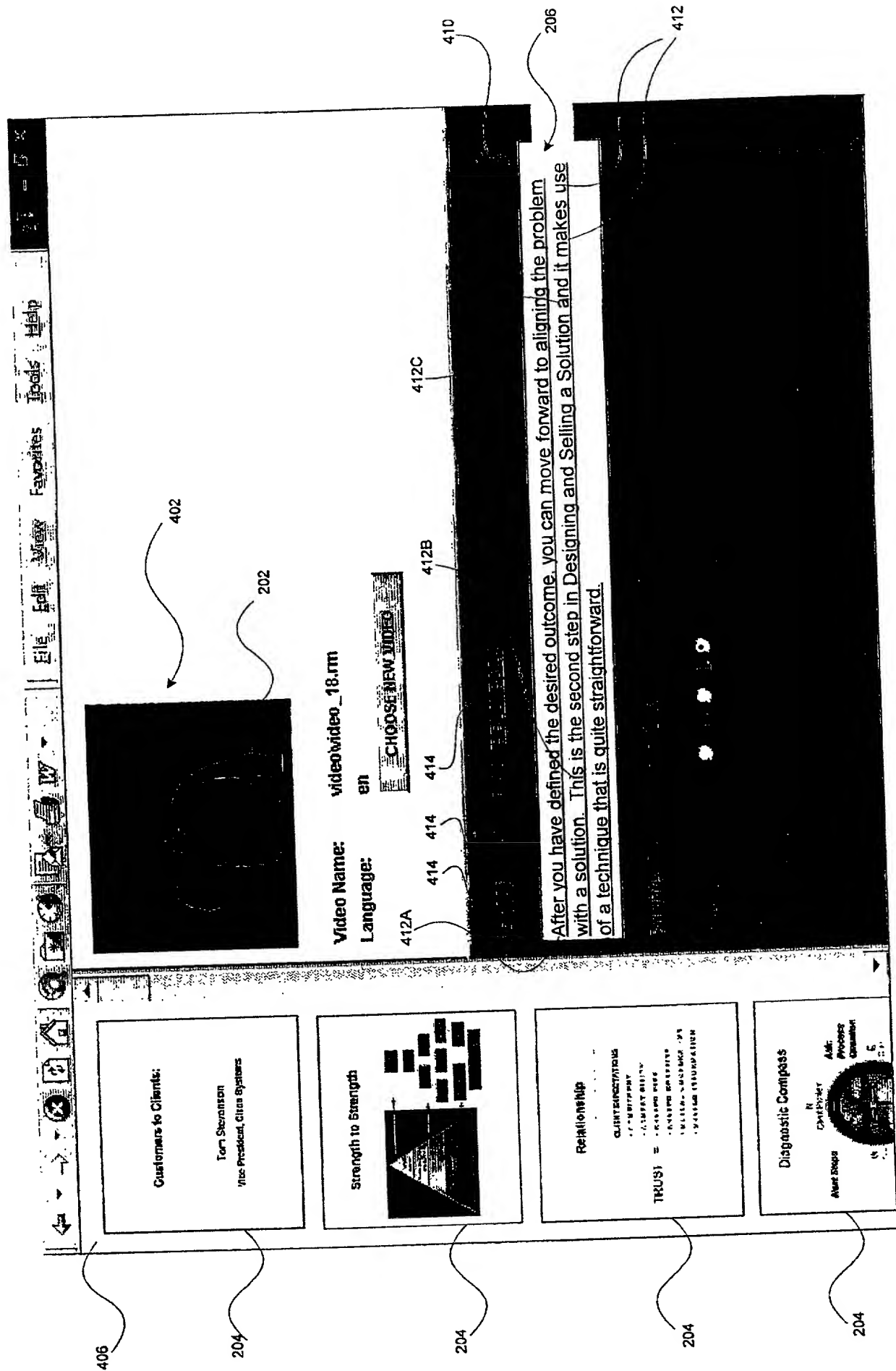


FIGURE 5A

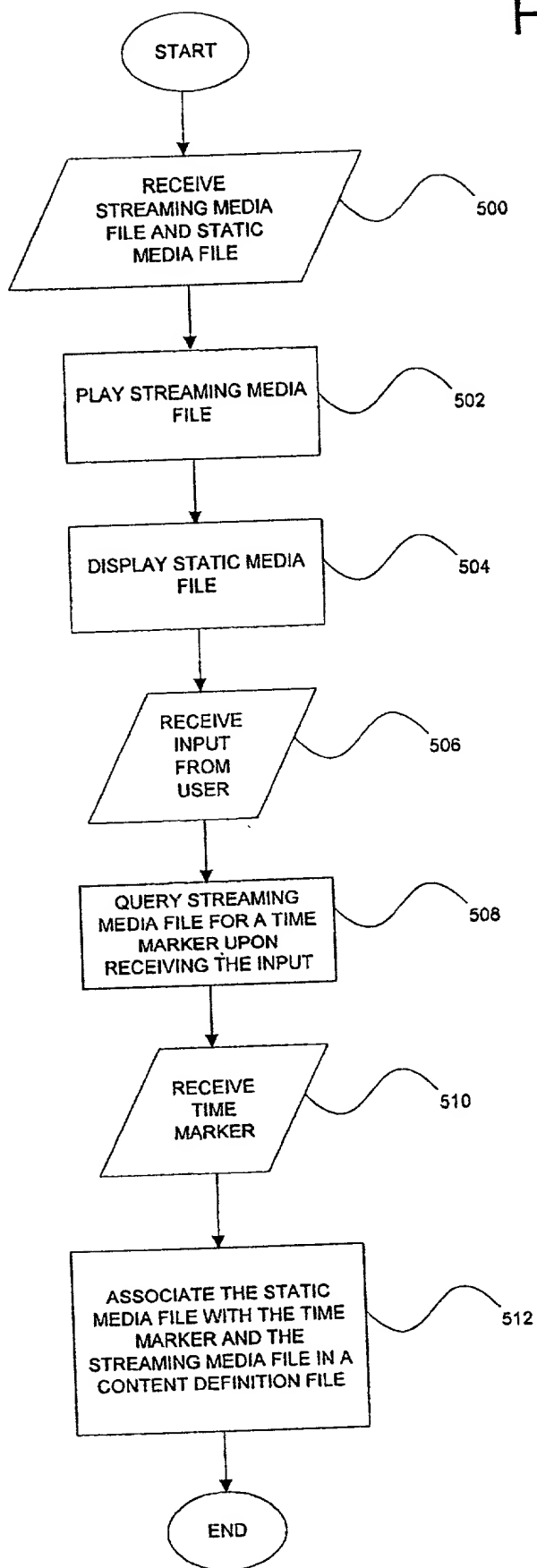


FIGURE 5B

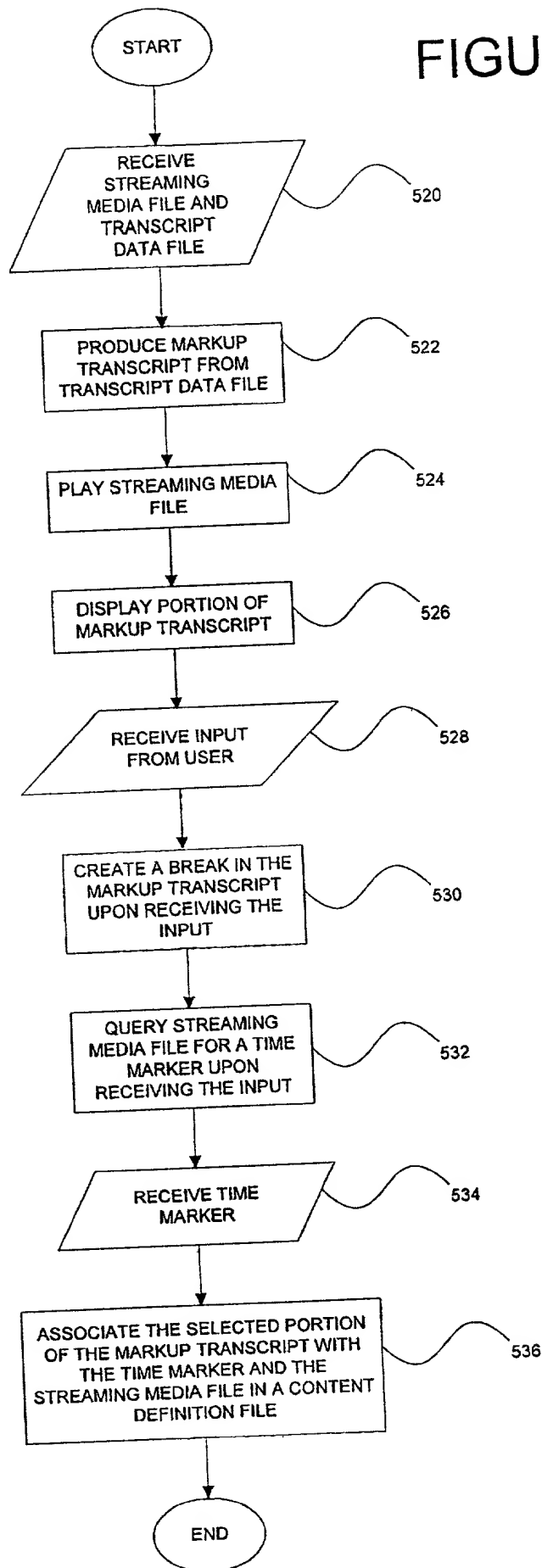


FIGURE 6A

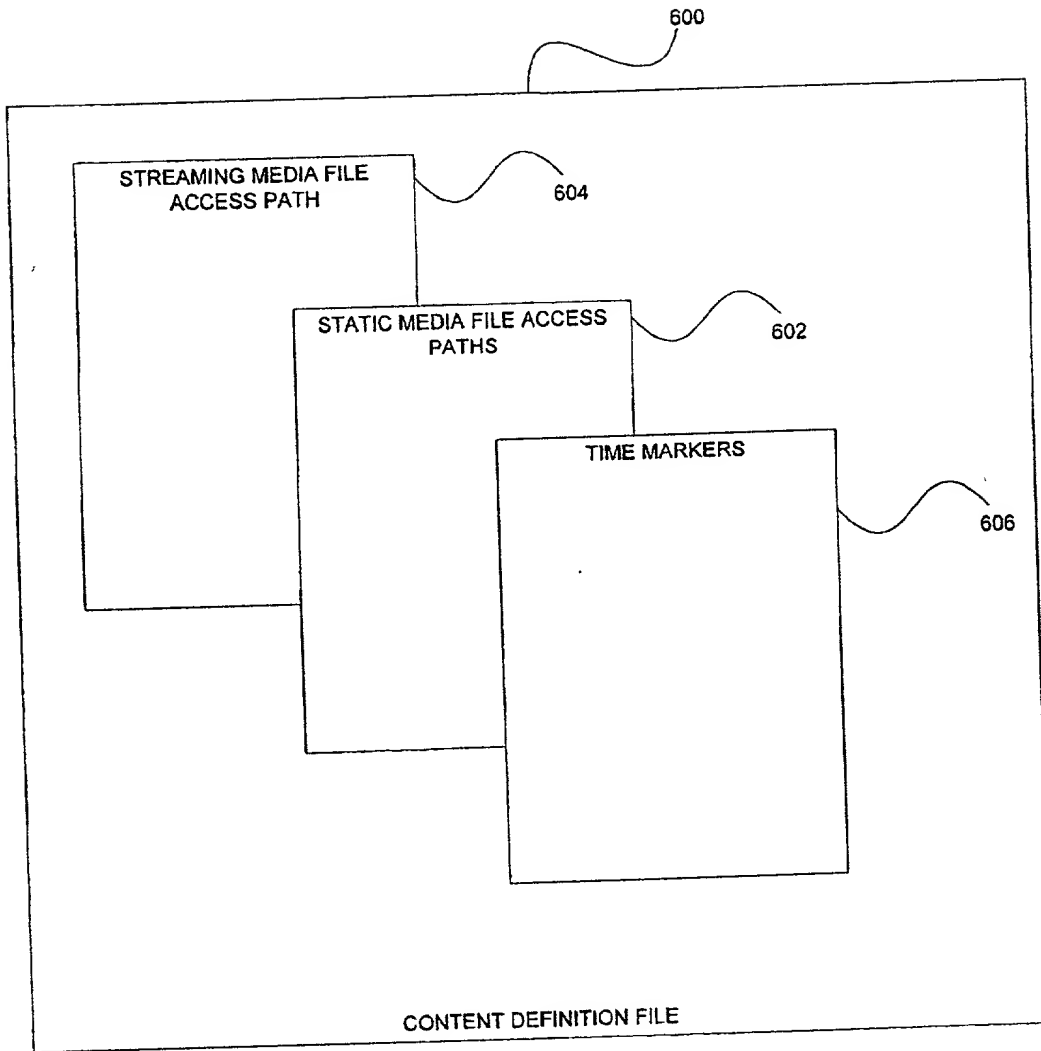
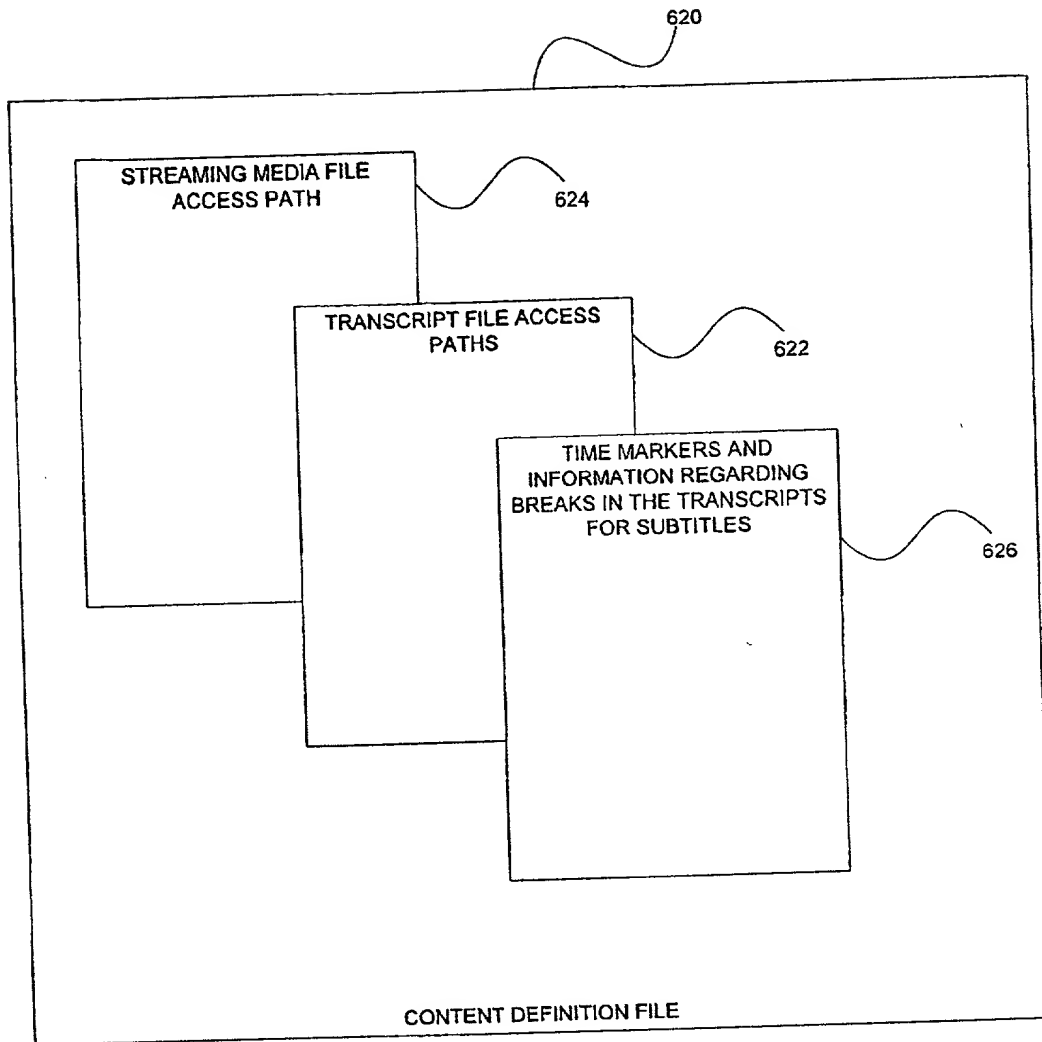


FIGURE 6B



DocId: 36469550

000001"00465960

202

STREAMING
MEDIA FILE

t = 00

01

02

03

04

05

06

07

08

09

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

STATIC
MEDIA FILES

706

204

SLIDE 00

706

706

204

706

SLIDE 01

706

204

SLIDE 02

SUBTITLES

704

TEXT 00

704

TEXT 01

704

TEXT 02

704

TEXT 03

FIGURE 7

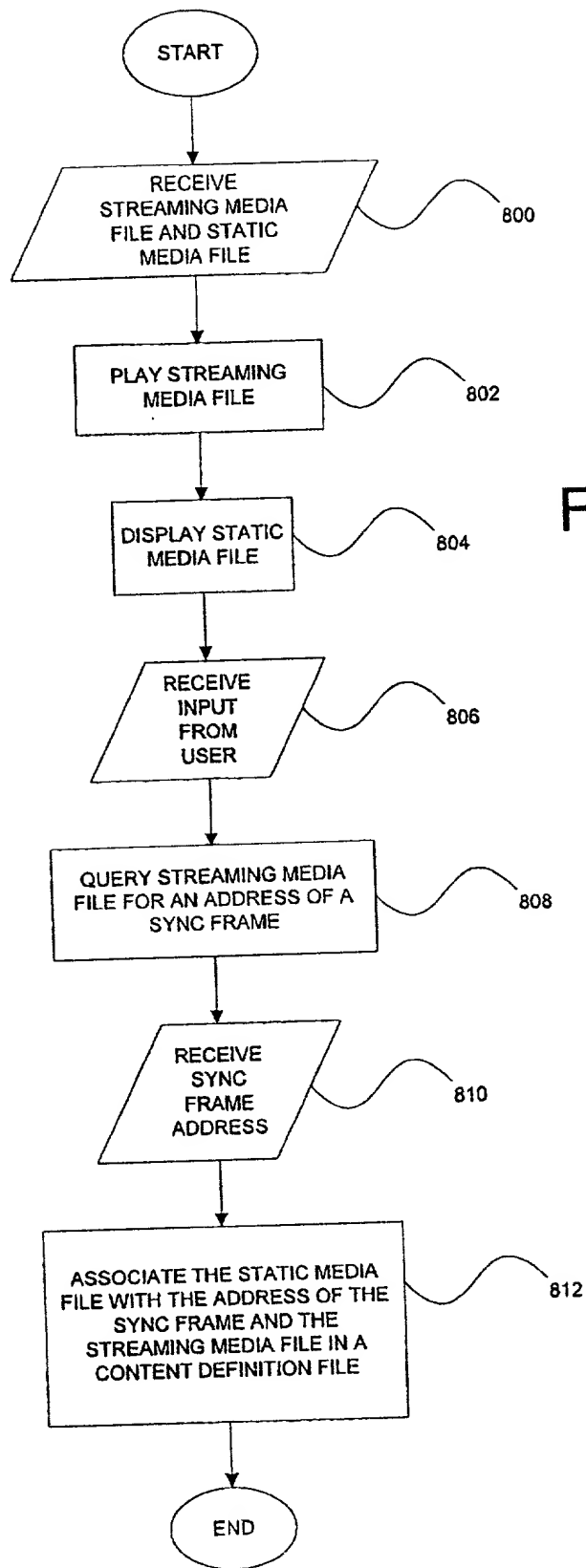
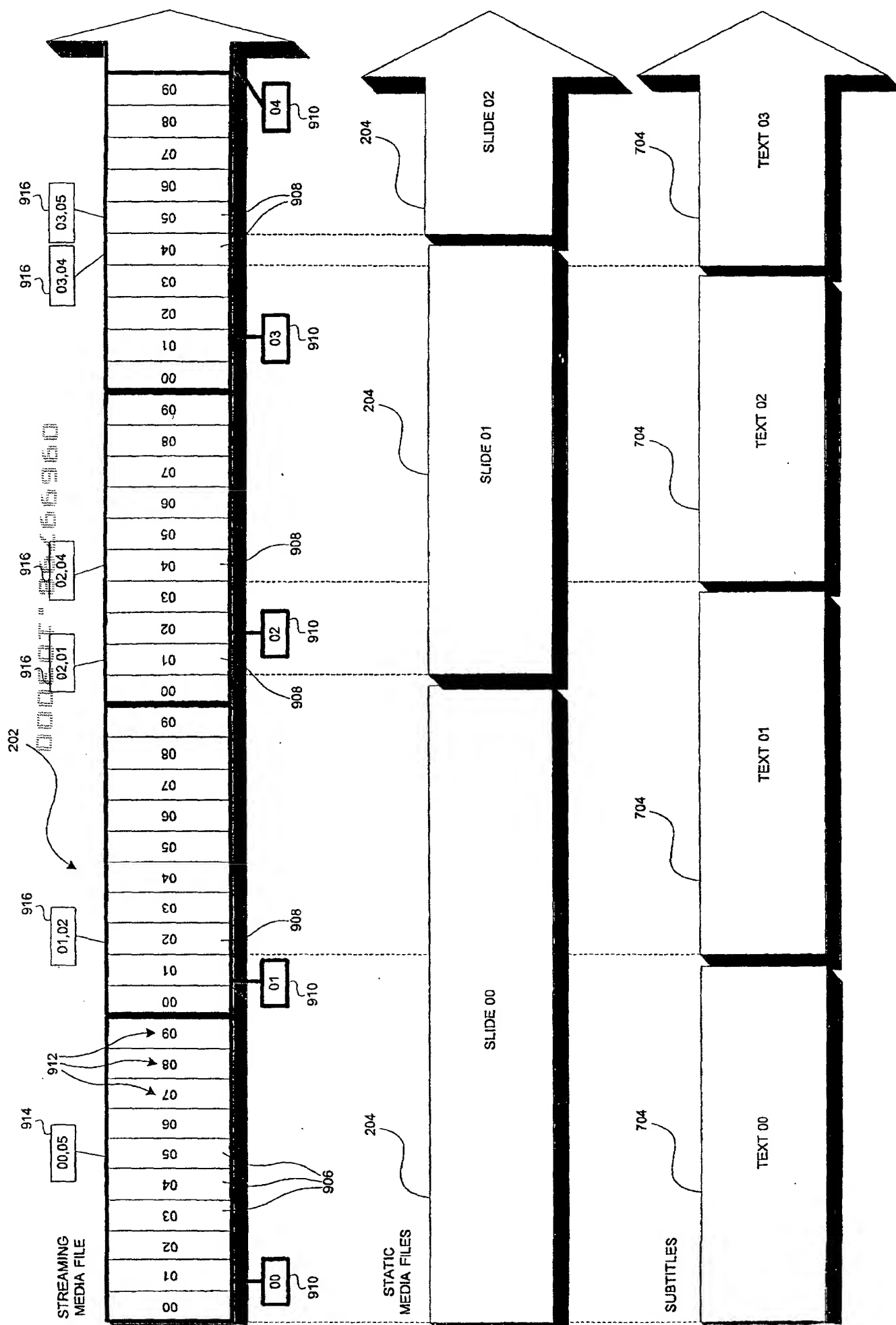


FIGURE 8



START

FIGURE 10

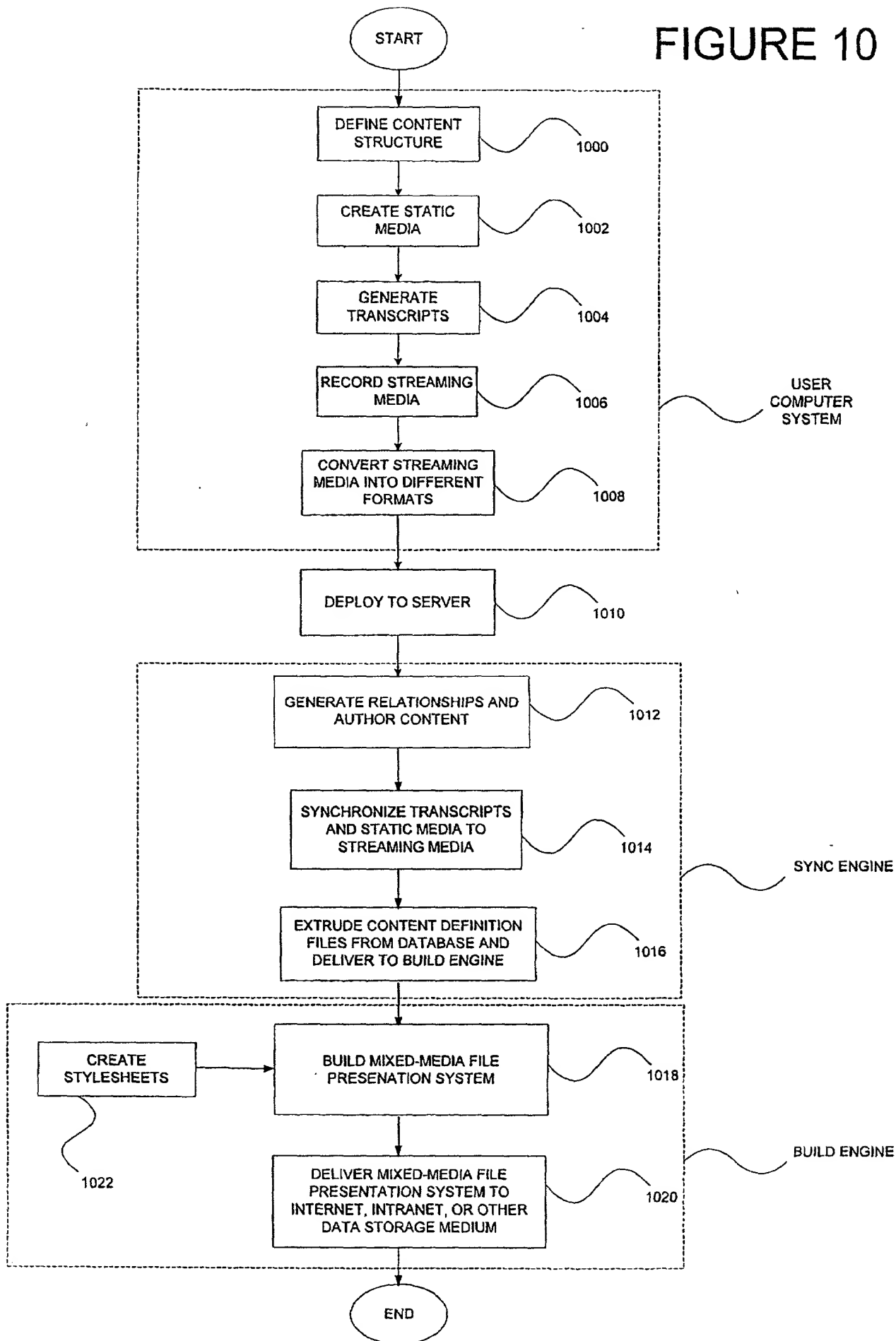
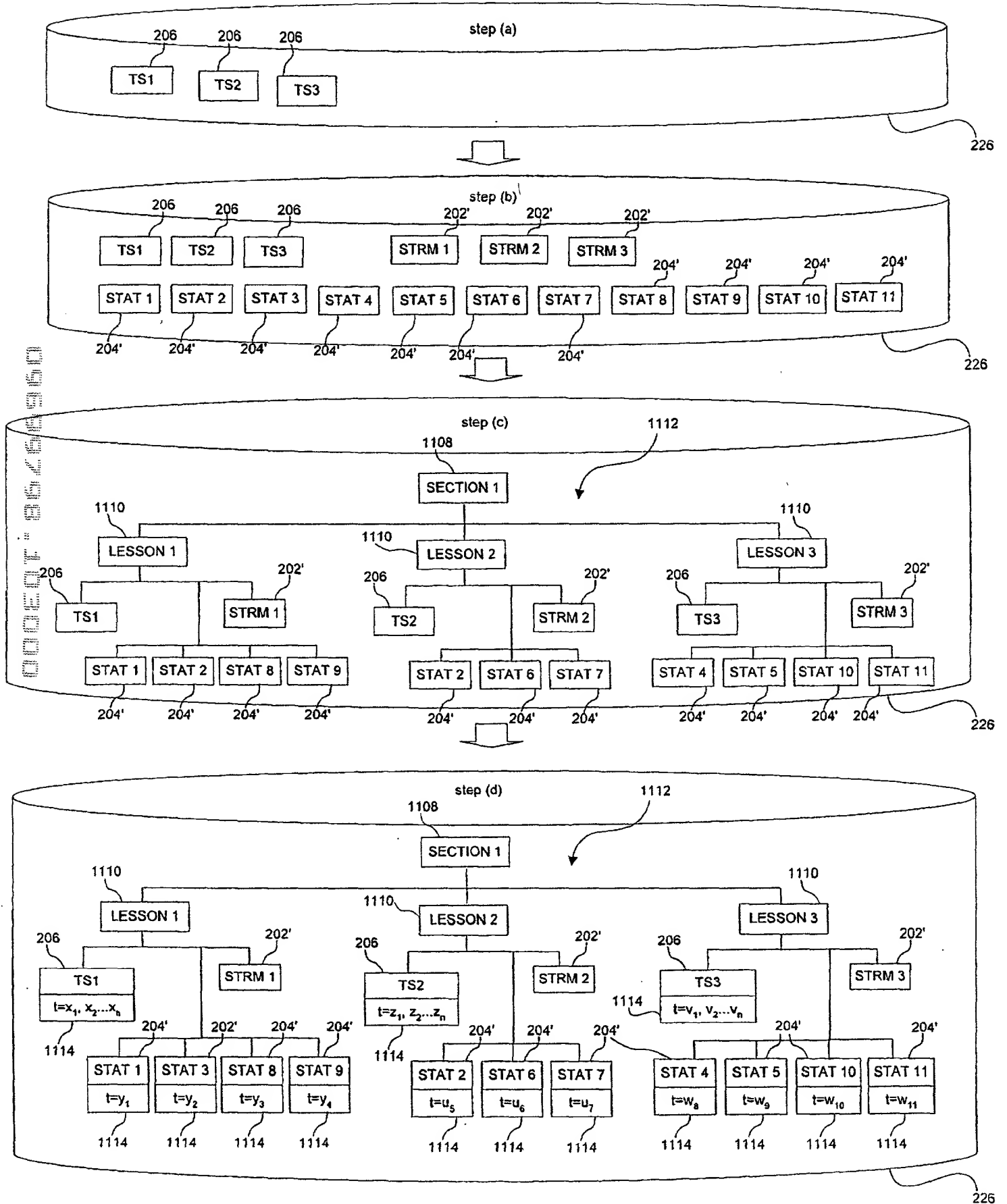


FIGURE 11



The diagram illustrates a computer system 1200. At the top, an external network configuration is shown: a SERVER 1230 is connected to an INTERNET/ISP (1228), which is connected to a LOCAL NETWORK 1222, which in turn is connected to a HOST 1224. A NETWORK LINK 1220 connects the LOCAL NETWORK 1222 to the computer system 1200. The computer system 1200 is enclosed in a large rectangle. Inside, a central BUS 1202 is connected via bidirectional arrows to several components: a PROCESSOR 1204, MAIN MEMORY 1206, ROM 1208, a COMMUNICATION INTERFACE 1218, and a STORAGE DEVICE 1210. Below the main system rectangle, three separate boxes represent peripheral devices: a DISPLAY 1212, an INPUT DEVICE 1214, and a CURSOR CONTROL 1216. These are connected to the BUS 1202 through a single vertical line that branches into three horizontal lines, each leading to one of the peripheral devices.

FIGURE 12

As a below named inventor, I hereby declare that:

I believe I am an original, first and joint inventor of the subject matter which is claimed and for which a patent is sought on the invention entitled METHODS AND APPARATUSES FOR SYNCHRONIZING MIXED-MEDIA DATA FILES; the specification of which is attached hereto;

I acknowledge the duty to disclose information which is material to patentability as defined in Title 37, Code of Federal Regulations, § 1.56;

POWER OF ATTORNEY: I hereby appoint the registrants of Knobbe, Martens, Olson & Bear, LLP, 620 Newport Center Drive, Sixteenth Floor, Newport Beach, California 92660, Telephone (949) 760-0404, **Customer No. 20,995**.

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful, false statements may jeopardize the validity of the application or any patent issued thereon.

Full name of first inventor: Timothy J. Smith

Inventor's signature

Date _____

Residence: 1335 Cristina Avenue
San Jose, California 95125
Citizenship: USA

Post Office Address: Same as above

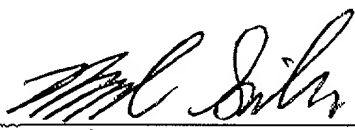
Page 2

Attorney's Docket No. HAYSCHR.002A

POWER OF ATTORNEY: I hereby appoint the registrants of Knobbe, Martens, Olson & Bear, LLP, 620 Newport Center Drive, Sixteenth Floor, Newport Beach, California 92660, Telephone (949) 760-0404, **Customer No. 20,995**, my attorney in fact for purposes of the above-referenced patent application only.

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful, false statements may jeopardize the validity of the application or any patent issued thereon.

Full name of inventor: Brad Silen

Inventor's signature 

Date 10/27/2000


Residence: #5 South Forty Dock
Sausalito, CA 94965

Citizenship: USA

Post Office Address: P.O. Box 1927
Sausalito, CA 94966

\\DOCS_SF\FILES\DOCS\TFS\TFS-6154.DOC
102700

Send Correspondence To:
KNOBBE, MARTENS, OLSON & BEAR, LLP
Customer No. 20,995


2 of 2